Computer Networks

Dr. Ali Sayyed

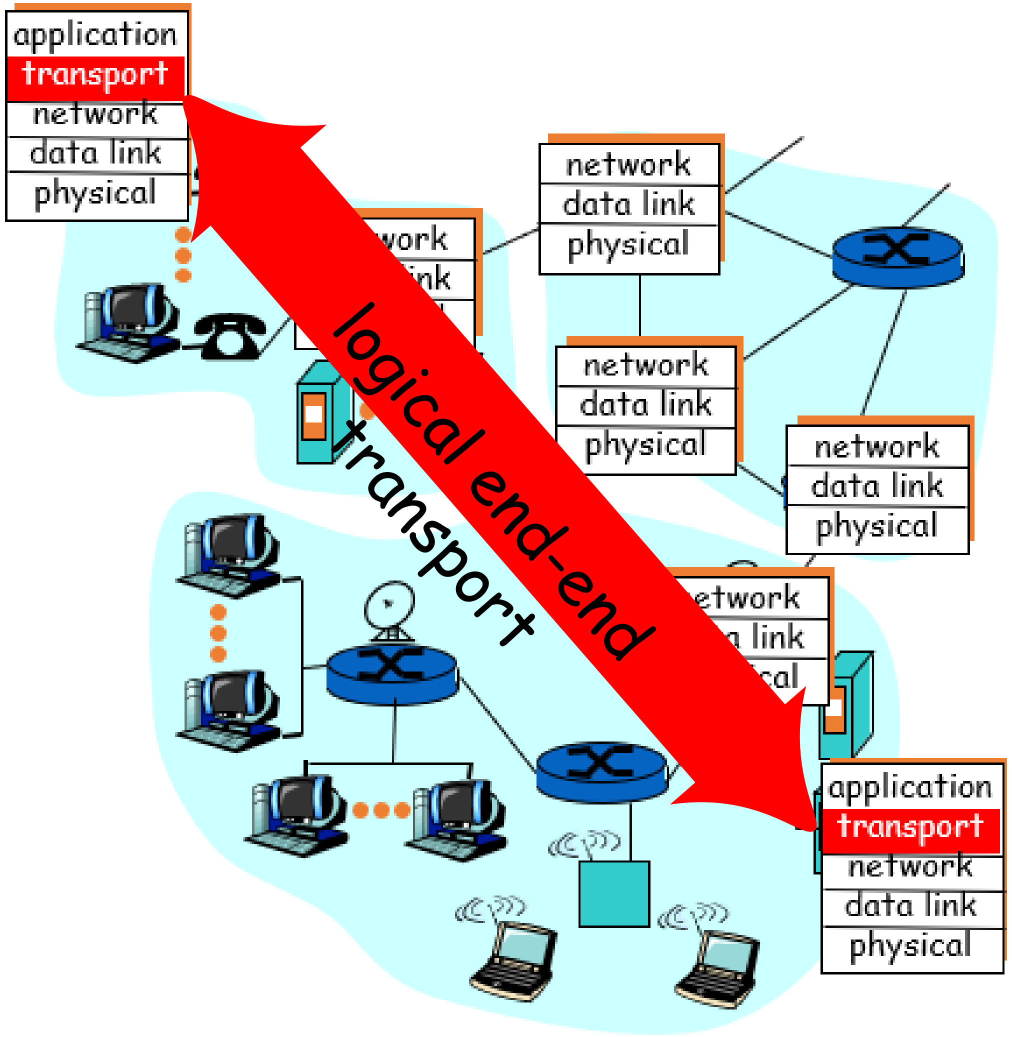
Department of Computer Science

National University of Computer & Emerging Sciences

Transport Layer

Transport Layer

The Transport layer is a true end-to-end layer, all the way from the source to the destination, providing a logical connection.



Layer 4 of the OSI model, also known as the transport layer, manages network traffic between hosts and end systems to ensure complete data transfers.

Transport Layer

- Some of the services provided by the transport layer are similar to those of the data link layer. However, the data link layer provides the services within a single network while the transport layer provides the services across an internetwork made up of many networks.
- Functions of Transport Layer
  - Service Point Addressing
  - Connection Control
  - Segmentation and Reassembling
  - Flow Control
  - Error Control
  - Congestion Control

Service Point Addressing

- Computers often run several programs at the same time.
- For this reason, source-to-destination delivery means delivery not only from one computer to the next but also from a specific process (running program) on one computer to a specific process (running program) on the other.
- The transport layer header must therefore include a type of address called a **service-point address** (or **port address**).
- The network layer gets each packet to the correct computer while the transport layer gets the entire message to the correct process on that computer.

Connection Control

It includes 2 types of connection control:

- **Connectionless Transport Layer :**
  - Each segment is considered as an independent packet and delivered to the transport layer at the destination machine.
- **Connection Oriented Transport Layer :**
  - A connection oriented transport layer makes a connection with the transport layer at the destination machine first before delivering the packets. After all the data are transferred, the connection is terminated.

TCP & UDP

- Transmission Control Protocol (TCP)
  - Connection oriented
- User Datagram Protocol (UDP)
  - Connectionless

User Datagram Protocol (UDP)

- The User Datagram Protocol (UDP) is called a connectionless and unreliable transport protocol.
- Delivery and duplication control not guaranteed
- UDP is a very simple protocol using a minimum of overhead. If a process wants to send a small message and does not care much about reliability, it can use UDP.
- Uses
  - Inward data collection
  - Request-Response
  - Real time applications, some routing protocols, network management

UDP Operation

- **Connectionless Services**
  - UDP provides a connectionless service. Each datagram sent by UDP is an independent datagram. There is no relationship between the different user datagrams even if they are coming from the same source and going to the same destination program. The user datagrams are not numbered.
- **Flow and Error Control**
  - UDP is a very simple, unreliable transport protocol. There is no flow control. There is no error control mechanism in UDP except for the 16 bit checksum. The lack of flow control and error control means that the process using UDP should provide these mechanisms.

TCP

- TCP is a reliable transport layer protocol.
- This means that an application program that delivers a stream of data to TCP relies on TCP to deliver the entire stream to the application program on the other end **in order**, **without error**, and **without any part lost or duplicated**.
- In TCP, connection-oriented transmission requires three phases:
  1. Connection establishment
  2. Data transfer
  3. Connection termination.

TCP Connection Establishment

- The connection establishment in TCP is called three way handshaking.
- Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections.
- Then a client send **SYN** message.
- In response, the server replies with a **SYN-ACK**.
- Finally, the client sends an **ACK** back to the server.
- In these steps, the connection parameter is decided and is acknowledged and as a result a full-duplex communication is established.

TCP Services - Segmentation and Reassembling

- A message is divided into transmittable segments, with each segment containing a sequence number.
- These sequence numbers enable the transport layer to **reassemble** the message correctly upon arriving at the destination and to identify and replace packets that were **lost** in transmission.
- The sequence number (random) of the first byte is chosen by the transmitter for the first packet.
- Acknowledgements (Acks) are sent with a sequence number
- Sequence number enable receiver to recognize **duplicates**

TCP Services – Checksum (Error control)

- Each TCP segment **includes a checksum** field which is used to check for a corrupted segment.
- If the segment is corrupted, it is discarded by the destination TCP and is considered as lost.
- TCP uses a **16-bit checksum** that is **mandatory** in every segment.
- The TCP checksum is a **weak** check by modern standards.
- However, the TCP checksum **catches most** of these simple errors.

TCP Services – Acknowledgment

- TCP uses acknowledgments to confirm the receipt of data segments.
- Control segments that carry no data but consume a sequence number are also acknowledged.
- ACK segments are never acknowledged.

TCP Services – Retransmission

- The heart of the error control mechanism is the retransmission of segments.
- When a segment is corrupted, lost, or delayed, it is retransmitted.
- A segment is retransmitted on two occasions:
  - Retransmission After RTO Expires
  - Retransmission After Three ACK

TCP Services – Retransmission After RTO Expires

- TCP maintains one retransmission time-out (RTO) timer for all outstanding (sent, but not acknowledged) segments.
- When the timer matures, the outstanding segment is retransmitted.
- Note that no time-out timer is set for a segment that carries only an acknowledgment.
- The value of RTO is dynamic in TCP and is updated based on the round-trip time (RTT) of segments.
- An RTT is the time needed for a segment to reach a destination and for an acknowledgment to be received.

TCP Services – Retransmission After Three ACK

- Sometimes, however, one segment is lost and the receiver receives so many out-of-order segments that they cannot be saved (limited buffer size). To avoid this situation, most implementations today follow the three-duplicate-ACKs rule and retransmit the missing segment immediately.
- If a single segment (say segment 100) in a stream is lost, then the receiver cannot acknowledge segment above 100 because it uses cumulative acks.
- Hence the receiver acknowledges segment 99 again on the receipt of segment 101 and onward.
- This duplicate acknowledgement is used as a signal for packet loss. That is, if the sender receives three duplicate acknowledgements, it retransmits the last unacknowledged packet.
- This feature is referred to as **fast retransmission**

TCP Services – Out-of-Order Segments

- When a segment is delayed, lost, or discarded, the segments following that segment arrive out of order.
- The out-of-order segments are stored temporarily and flag them as out-of-order segments until the missing segment arrives.
- Note, however, that the out-of-order segments are not delivered to the process (running program).
- TCP guarantees that data are delivered to the process (running program) in order.

TCP Services – Retransmissions

- **Positive Acknowledgement (ACK)**
  - The receiver explicitly notifies the sender about which segments were received correctly. Positive Acknowledgement therefore also implicitly informs the sender which packets were not received and provides detail on packets which need to be retransmitted.
- **Negative Acknowledgment (NACK)**
  - The receiver explicitly notifies the sender which packets, messages, or segments were received incorrectly and thus may need to be retransmitted

Flow Control

- Like the data link layer, the transport layer is responsible for flow control. However, **flow control at this layer is performed end to end** rather than across a single link.
- Flow Control basically means that TCP will ensure that a sender is not overwhelming a receiver by sending packets faster than it can consume.
- Flow control is essential in heterogenous networks.
- In each ACK, the receiver specifies a window size, the amount of data that it is willing to buffer for the connection.
- When a receiver advertises a window size of 0, the sender stops sending data.

Flow Control - The persist timer

- There's still one problem, though. After the receiver advertises a zero window, if it does send another ack message to the sender but it is lost, it will never know when it can start sending data again. There will be a deadlock situation, where the receiver is waiting for more data, and the sender is waiting for a message saying it can start sending data again.
- To solve this problem, when TCP receives a zero-window message it starts the persist timer, that will periodically send a small packet to the receiver (usually called WindowProbe), so it has a chance to advertise a nonzero window size.

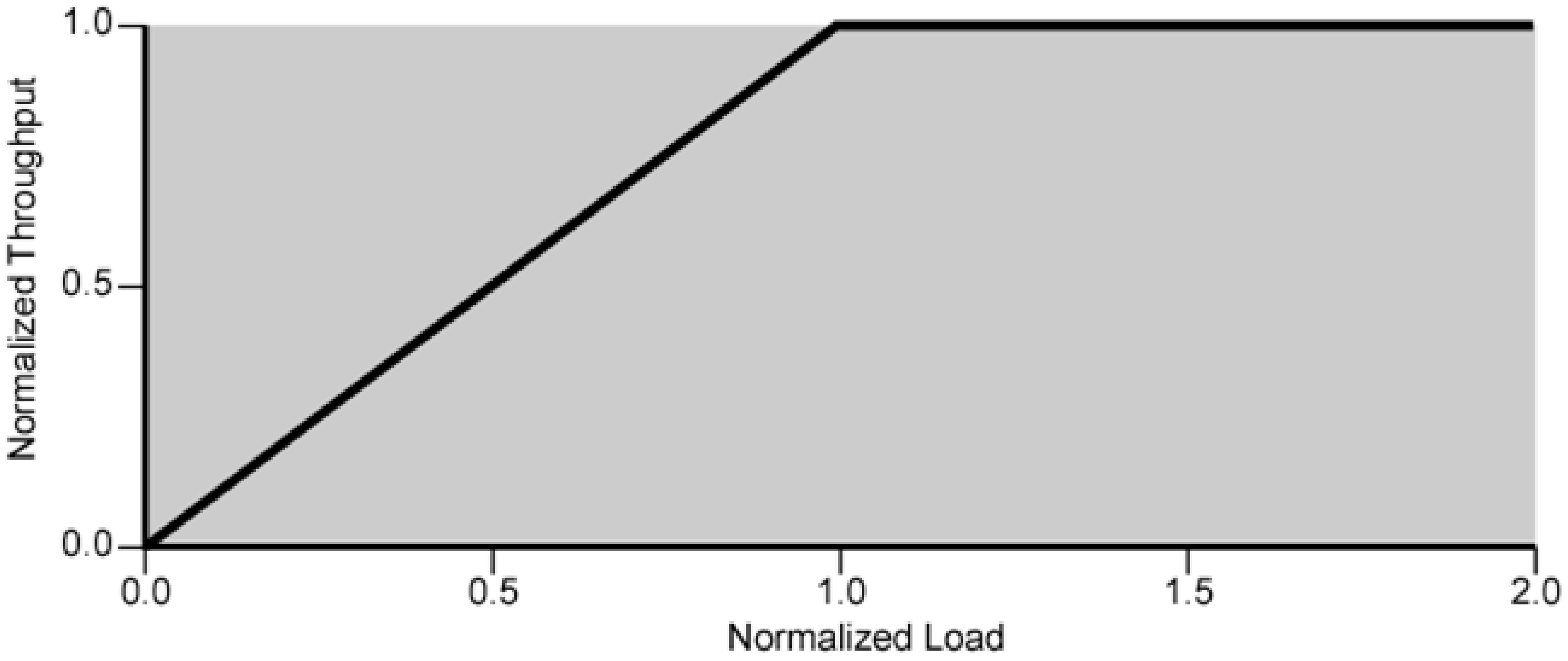
What Is Congestion?

- Congestion occurs when the number of packets being transmitted through the network approaches the packet handling capacity of the network
- Congestion control aims to keep number of packets below level at which performance falls off dramatically
- Data network is a network of queues
- Generally, 80% utilization is critical

Effects of Congestion

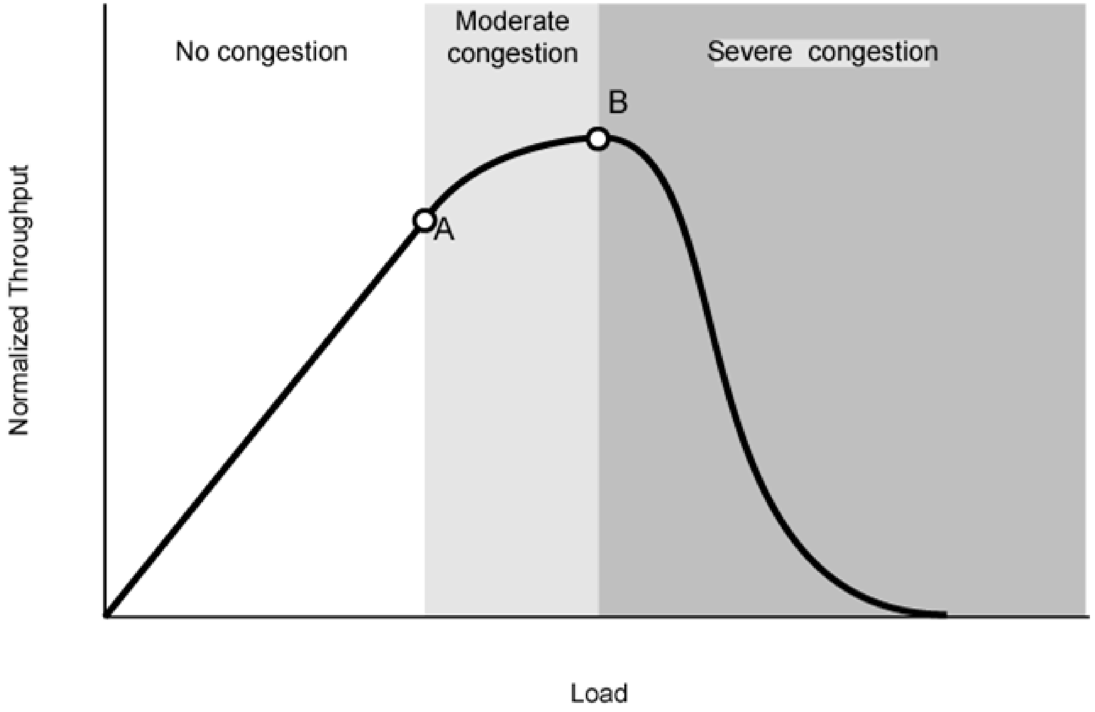
- Packets arriving are stored at input buffers
- Routing decision made
- Packet moves to output buffer
- Packets queued for output transmitted as fast as possible
- If packets arrive too fast to be routed, or to be output, buffers will fill
- Can discard packets
- While network congestion is usually temporary, it can cause inconvenient network problems that can affect performance, such as packet loss, and latency, as well as a decrease in throughput.

Ideal Network Utilization



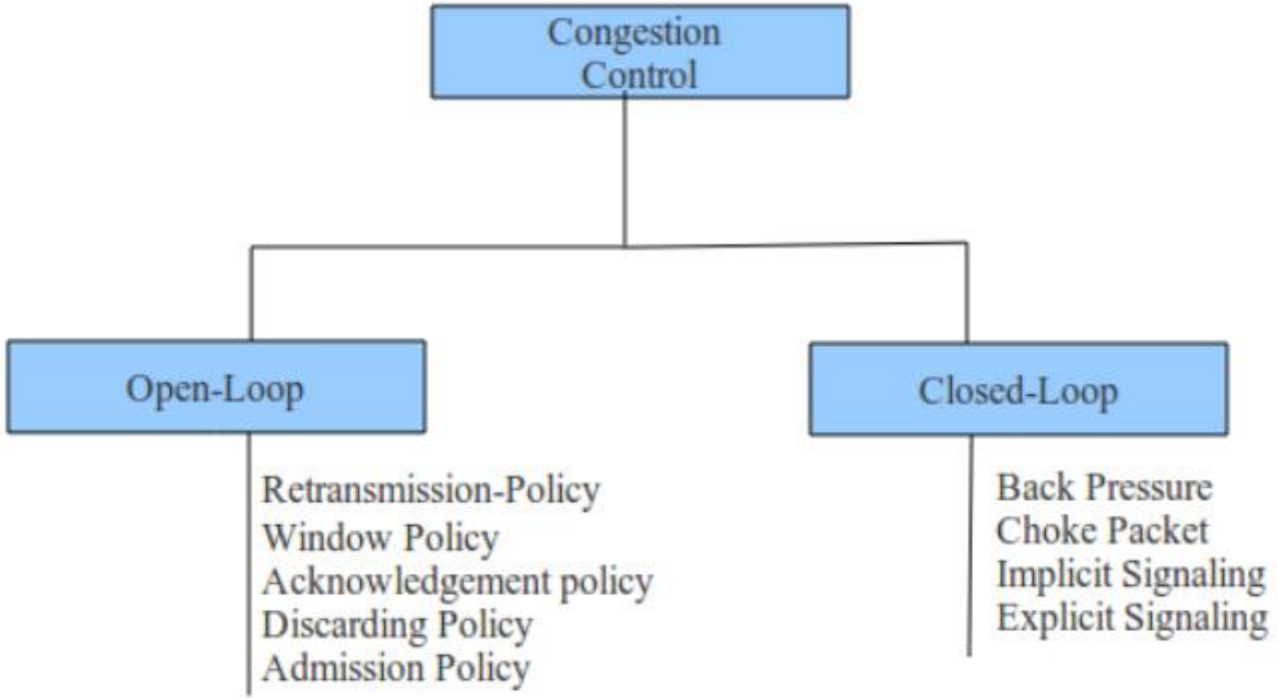
Practical Performance

- Ideal assumes infinite buffers and no overhead
- Buffers are finite
- Overheads occur in exchanging congestion control messages



Congestion Control Techniques

- Congestion control refers to the techniques used to control or prevent congestion. Congestion control techniques can be broadly classified into two categories:



Open Loop Congestion Control

- Open loop congestion control policies are applied to prevent congestion before it happens. The congestion control is handled either by the source or the destination.

Retransmission Policy

- Retransmission is sometimes unavoidable. If the sender feels that a sent packet is lost or corrupted, the packet needs to be retransmitted. Retransmission in general may increase congestion in the network. However, a good retransmission policy can prevent congestion. The retransmission policy and the retransmission timers must be designed to optimize efficiency and at the same time prevent congestion.

Open Loop Congestion Control

Retransmission Protocol Policy

- The Selective Repeat window is better than the Go-Back-N window for congestion control.

Discarding Policy

- A good discarding policy by the routers may prevent congestion and at the same time may not harm the integrity of the transmission. For example, in audio transmission, if the policy is to discard less sensitive packets when congestion is likely to happen, the quality of sound is still preserved and congestion is prevented or alleviated.

Open Loop Congestion Control

Acknowledgment Policy

- The acknowledgment policy imposed by the receiver may also affect congestion. If the receiver does not acknowledge every packet it receives, it may slow down the sender and help prevent congestion.
- Several approaches are used in this case. A receiver may decide to acknowledge only N packets at a time. We need to know that the acknowledgments are also part of the load in a network.
- Sending fewer acknowledgments means imposing less load on the network.

Open Loop Congestion Control

Admission Policy

An admission policy, which is a quality-of-service mechanism, can also prevent congestion in virtual-circuit networks. Switches in a flow first check the resource requirement of a connection before admitting it to the network. A router can deny establishing a virtual circuit connection if there is congestion in the network or if there is a possibility of future congestion

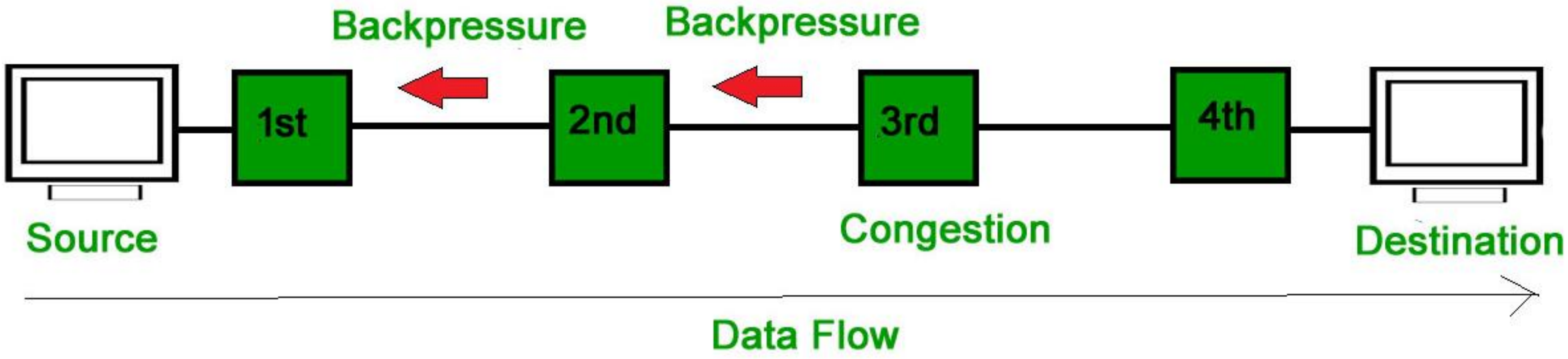
Closed Loop Congestion Control

Closed-loop congestion control mechanisms try to alleviate congestion after it happens.

Several mechanisms have been used by different protocols. We describe a few of them here.

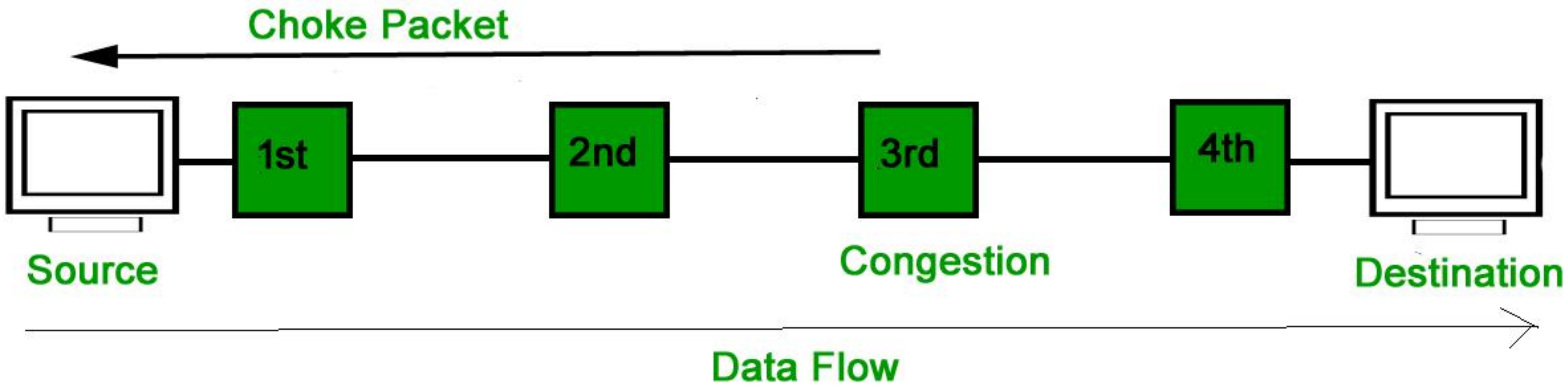
Backpressure

- The technique of backpressure refers to a congestion control mechanism in which a congested node stops receiving data from the immediate upstream node or nodes.
- This may cause the upstream node or nodes to become congested, and they, in turn, reject data from their upstream nodes or nodes. And so on.
- Propagates back to source



Choke Packet Technique

- Generated at congested node and Sent to source node
- A choke packet is a packet sent by a node to the source to inform it of congestion.
- Note the difference between the backpressure and choke packet methods.
- In the choke packet method, the warning is from the node, which has encountered congestion, to the source station directly. The intermediate nodes through which the packet has traveled are not warned.



Implicit and Explicit Congestion Signaling

- **Implicit Congestion Signaling**
  - No specific communication regarding the congestion
  - Transmission delay may increase
  - Packet may be discarded, ACK may not be received
  - Source can detect these as implicit indications of congestion
  - Useful on connectionless (datagram) networks

- **Explicit Congestion Signaling**
  - Network alerts end systems of increasing congestion
  - End systems take steps to reduce offered load

Implicit and Explicit Congestion Signaling

- **Explicit Congestion Signaling**
  - **Forward Signaling** : In forward signaling, a signal is sent in the direction of the congestion. The destination is warned about congestion. The receiver in this case adopt policies to prevent further congestion.

  - **Backward Signaling** : In backward signaling, a signal is sent in the opposite direction of the congestion. The source is warned about congestion and it needs to slow down.